

<b>Framework</b>	Cucumber BDD framework
<b>Design Pattern</b>	Singleton, Page Object model (POM), <b>POJO</b>
<b>Languages</b>	Java OOP language, Gherkin, HTML, <b>SQL</b>
<b>Testing Tools</b>	Selenium WebDriver, Junit, Cucumber, TestNG, <b>Apache POI, Selenium Grid, JDBC, RestAssured</b>
<b>Build Management Tool</b>	Maven
<b>CI/CD tool</b>	<b>Jenkins</b>
<b>Project Methodology</b>	Waterfall, Agile, Agile-Scrum
<b>Bug Tracking / Project management tool</b>	Jira-Xray, (Jira)
<b>IDEs</b>	IntelliJ, (Eclipse), <b>SQL Developer, (DBeaver), Postman</b>
<b>Version Control System</b>	Git, GitHub, (Gitlab)
<b>Browsers</b>	Google Chrome, Firefox, Internet Explore, Edge, Safari
<b>Operating System</b>	MacOS, Microsoft Windows
<b>Testing types</b>	Dynamic, System, Functional, UI/FrontEnd, DataBase, API (backEnd), Smoke, Regression,, cross-browser, parallel testing

# Cloud Storage Domain

Automatic data **backup** for servers & computers

Secure, encrypted cloud

manage all types of files, photos, and videos

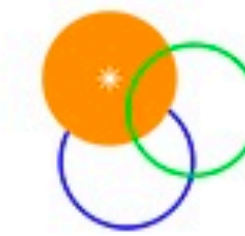
Remote file access from any device

Team collaboration

Communication



Google Drive



Codebase India



iCloud



MEGA



OneDrive

**Company names:** TryCloud



**What does this company do?**

This is a **software development** company

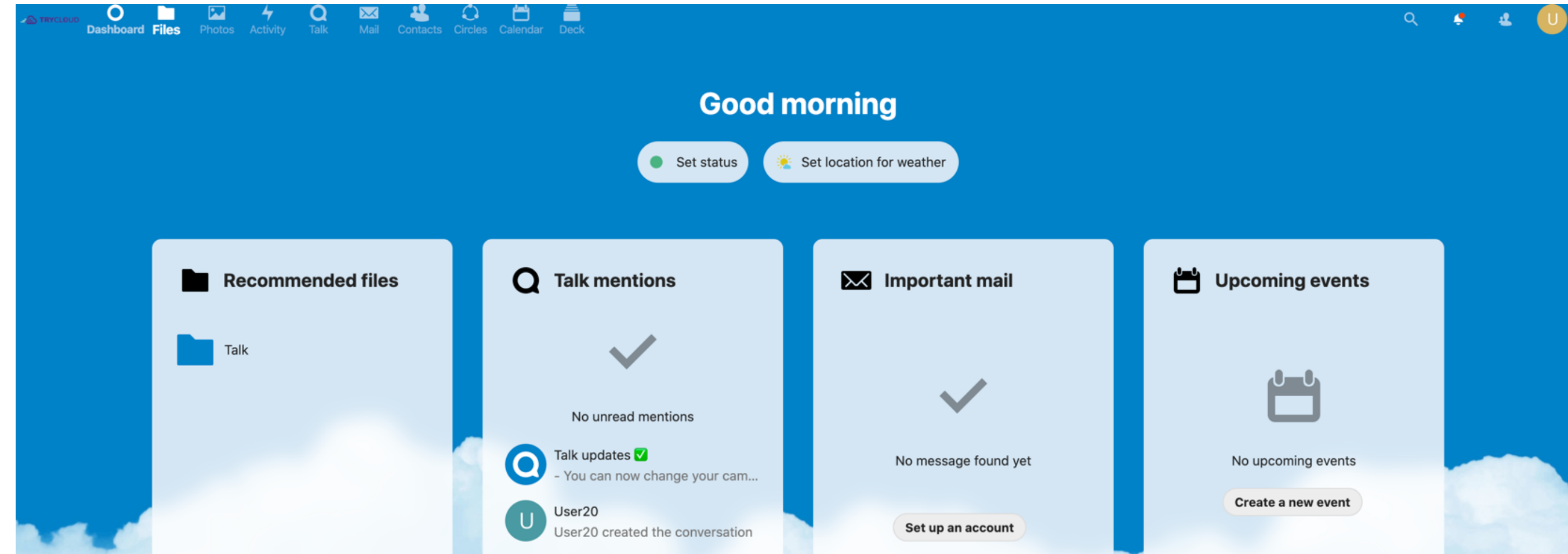
To build a Secure, data storage software

Along with team collaboration features

## Describe the TryCloud project?

This project is a cloud storage web-based app

Features are:



Safe and secure **file synchronization** / real-time working on sharing files through cloud server

User **contact management**

**Communication** - text / audio / video chat

Keyword **search** file & All types of files like: doc/ppt/xls/csv/psd/ai

**Customizable dashboards** and wallboards

**Calendar** & event creation

Customizable **workflows**

## Assignment Description

Each group will create a BDD Cucumber framework to test the TryCloud company's cloud storage web-based application in this assignment. Each student at least automates one user story.

**The framework will have:** Java

Maven

Selenium WebDriver, Junit, Cucumber

Design patterns - POM, Singleton

Git/Github

**Application information:**

Credentials link: [click here](#)

**Note:** Due to the time concern, agile-meetings practice & proof of manual testing execution is optional for this project.



## Assignment to-do list:

### 1. Each group creates a Cucumber-BDD Framework

- Conduct a meeting with the team to SET UP the project [look at the next slide for how to set up a project]
- One person is responsible to set up the project & push it to GitHub AND **add** all the **other** team members **as collaborators**
- All the team members accept the invitation & clone the project

### 2. Perform UI/FrontEnd automation

- Create one features file per story
- Create PR

**Note:** You will practice performing **Smoke & Regression testings** from this framework after learning Jenkins & Selenium Grid topics.

## Project set up

1. Create a maven project.

Group-id: **com.trycloud**

Artifact-id: **trycloudAutomation**

2. **Add** all the required **dependencies** for automation to the **pom file**.

3. Under **src/test/java/** create following packages:

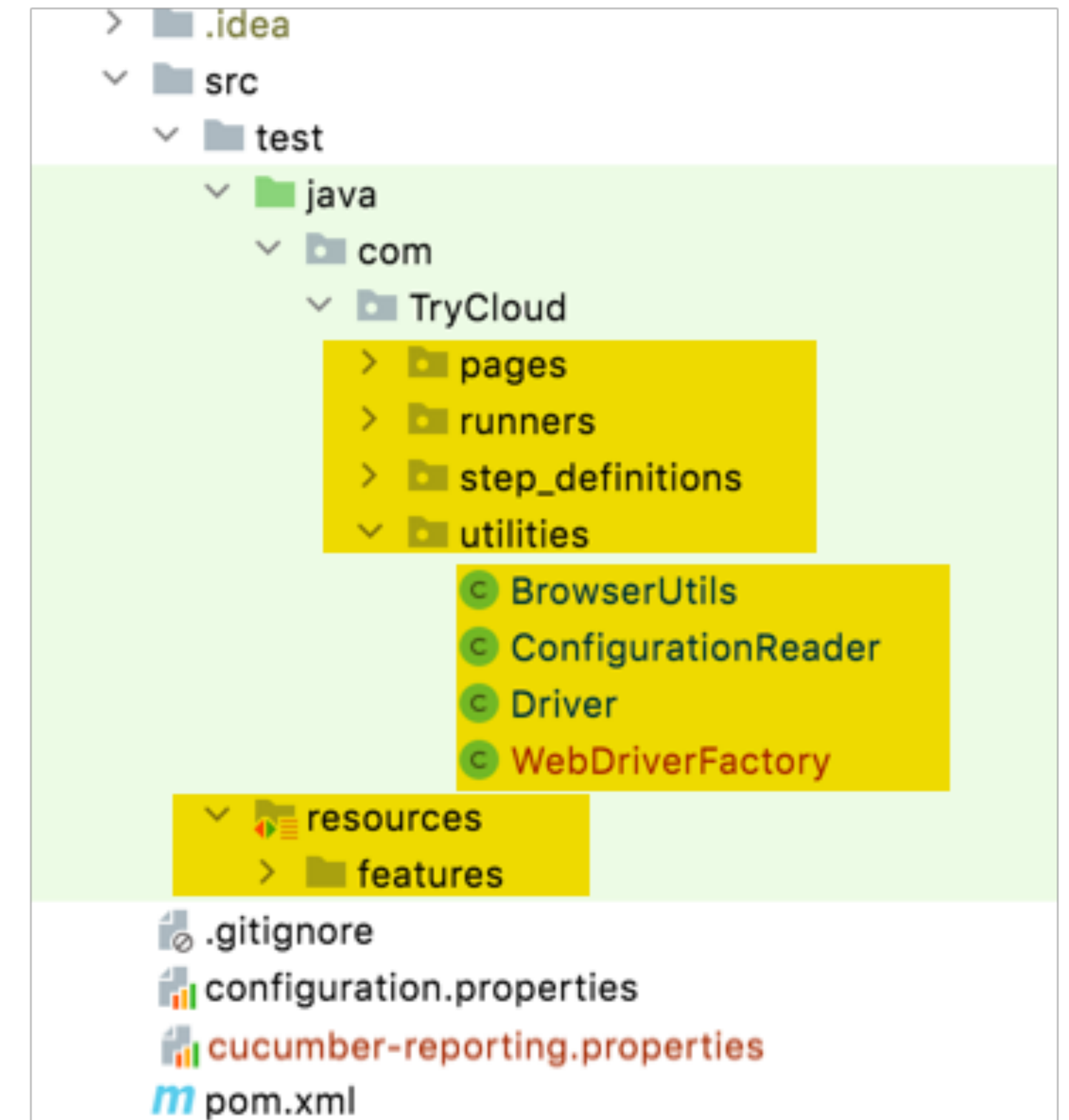
- a. **com.trycloud.pages**
- b. **com.trycloud.runner**
- c. **com.trycloud.step\_definitions**
- d. **com.trycloud.utilities**

4. Create “**resources**” directory and “**features**” package

5. Under the project folder add file **configuration.properties**  
add file **cucumber-reporting.properties**  
add file **.gitignore**

6. **Create feature file** based on the given user stories & Acs

7. **Create classes** under the step-definitions to **automate** the scenario or scenario outline



1. **Feature:** As a user, I should be able to log in.

**Scenario Outline:** Verify login with valid credentials

**Given** user on the login page <http://qa3.trycloud.net/index.php/login?clear=1>

**When** user use username "<username>" and passcode "<password>"

**And** user click the login button

**Then** verify the user should be at the dashboard page

**Examples:**

username	password
user7	Userpass123
user34	Userpass123
user99	Userpass123



2. **Feature:** As a user, I should be able to log in.

**Scenario Outline:** Verify user login **fail with invalid** credentials

**Given** user on the login page <http://qa3.trycloud.net/index.php/login?clear=1>

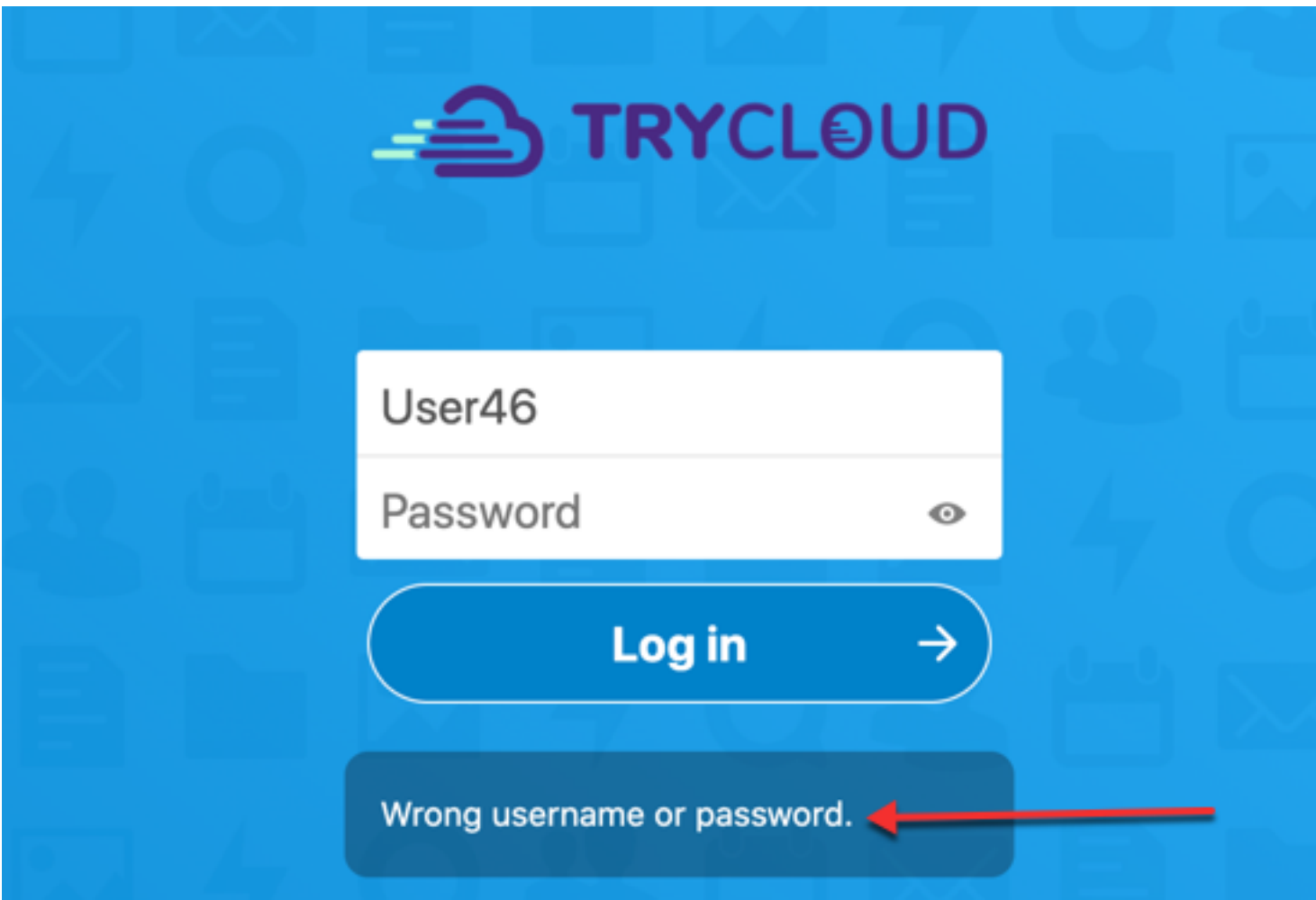
**When** user enter **invalid** "<username>" and "<password>"

**And** user click the login button

**Then** verify "<message>" message should be displayed

**Examples:**

username	password	message
User9	Wrong	Wrong username or password.
Wrong	Userpass123	Wrong username or password.
Wrong	Wrong	Wrong username or password.



3. **Story:** As a user, I should be accessing all the main modules of the app.

Feature: As a user, I should be accessing all the main modules of the app.

**Scenario:** Verify users accessing all the main modules of the app.

Given user on the login page

When the users log in with valid credentials

Then Verify the user see the following modules:

- | **Dashboard** |
- | **Files** |
- | **Photos** |
- | **Activity** |
- | **Talk.** |
- | **Contacts** |
- | **Circles** |
- | **Calendar** |
- | **Deck** |



4. **Story:** As a user, I should be able to access to **Files** module.

**Feature:** As a user, I should be able to access to Files module.

**Scenario:** verify users can access to Files module

Given user on the dashboard page

When the user clicks the "Files" module

Then verify the page title is "Files - Trycloud QA."

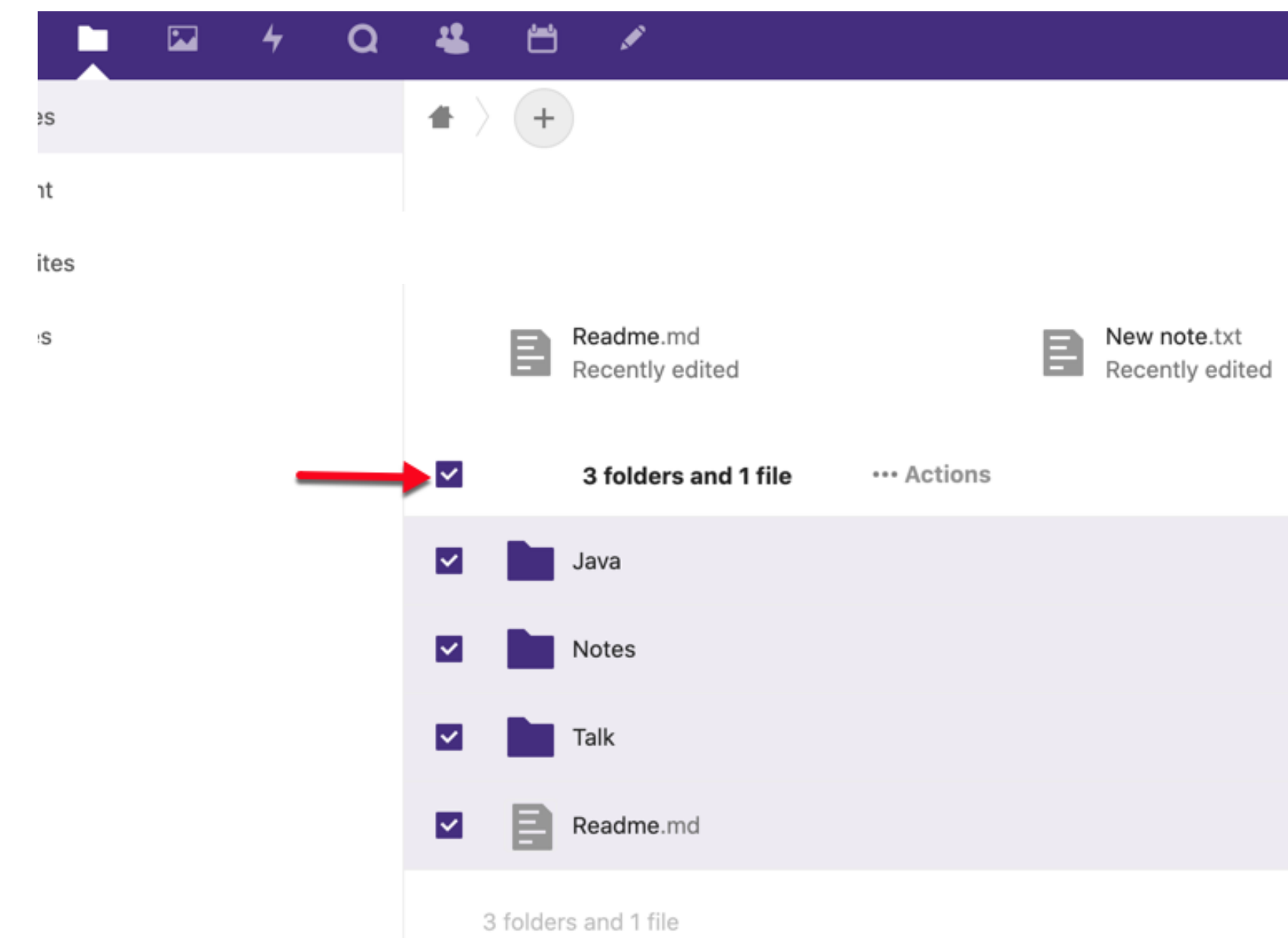
**Scenario:** verify users can select all the uploaded files from the page

Given user on the dashboard page

When the user clicks the "Files" module

And user click the top-left checkbox of the table

Then verify all the files are selected



5. **Story:** As a user, I should be able to add file to favorites.

Feature: As a user, I should be able to access to Files module - Favorites button

**Scenario:** Verify users to add files to Favorites

Given user on the dashboard page

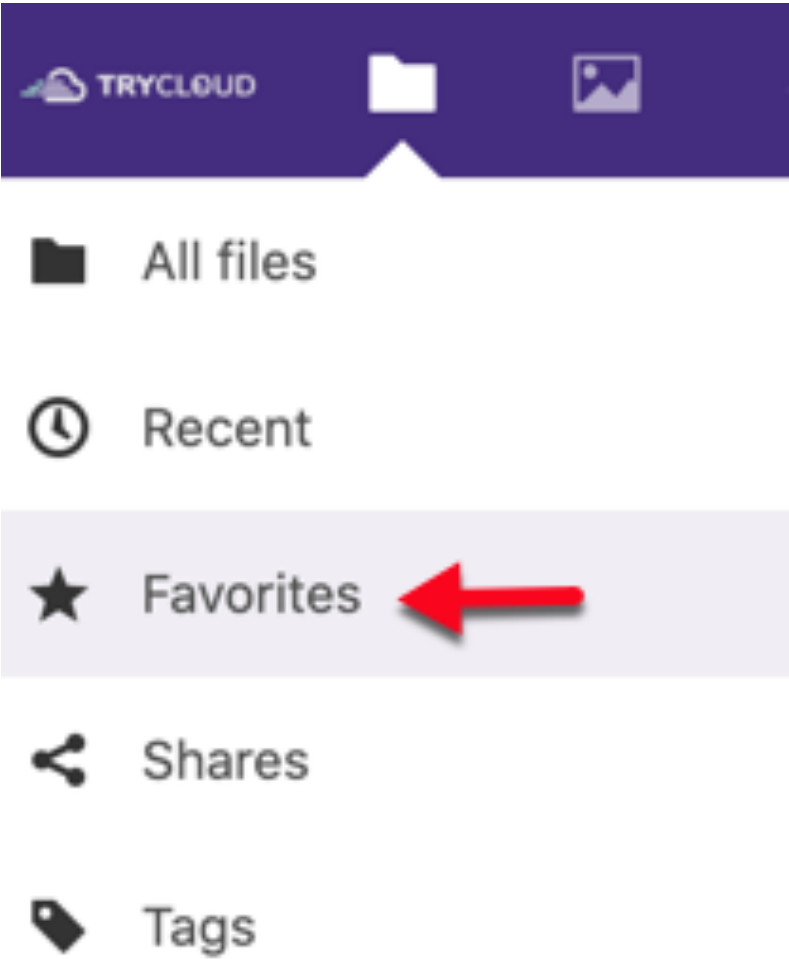
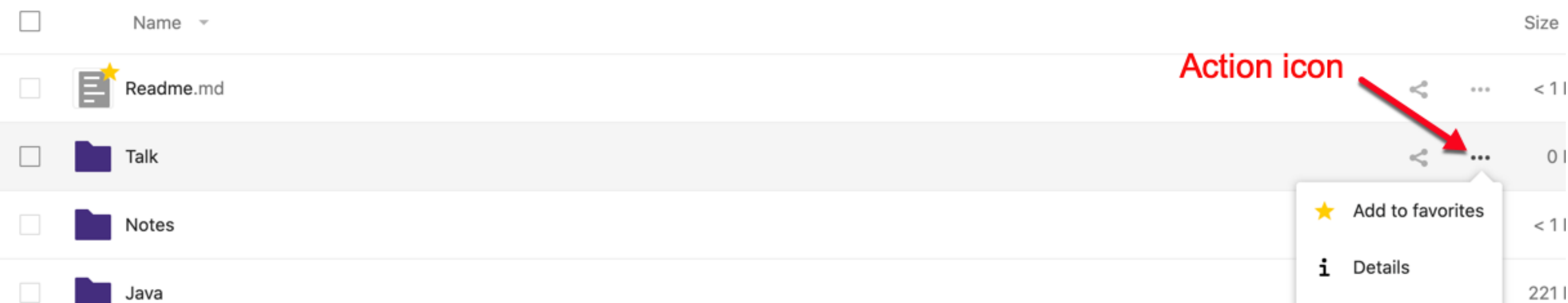
When the user clicks the "Files" module

When the user clicks action-icon from any file on the page

And user choose the "Add to favorites" option

And user click the "Favorites" sub-module on the left side

Then Verify the chosen file is listed on the table



6. **Story:** As a user, I should be able to edit favorites files.

Feature: As a user, I should be able to remove files from favorites and upload a file directly

**Scenario:** verify users to remove files to Favorites

Given user on the dashboard page

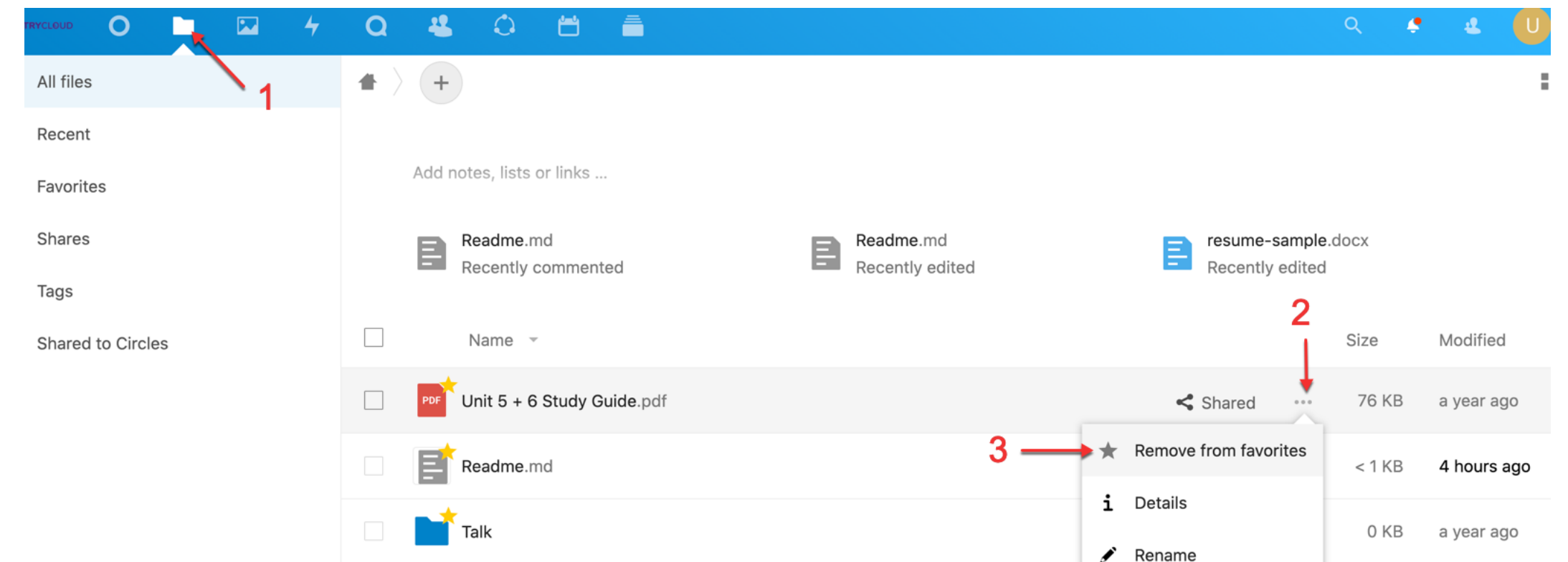
When the user clicks the "Files" module

When the users click action-icon from any file on the page to remove

And user choose the "Remove from favorites" option

And user click the "Favorites" sub-module on the left side

Then Verify that the file is removed from the Favorites sub-module's table



**Scenario:** verify users to upload a file from Files

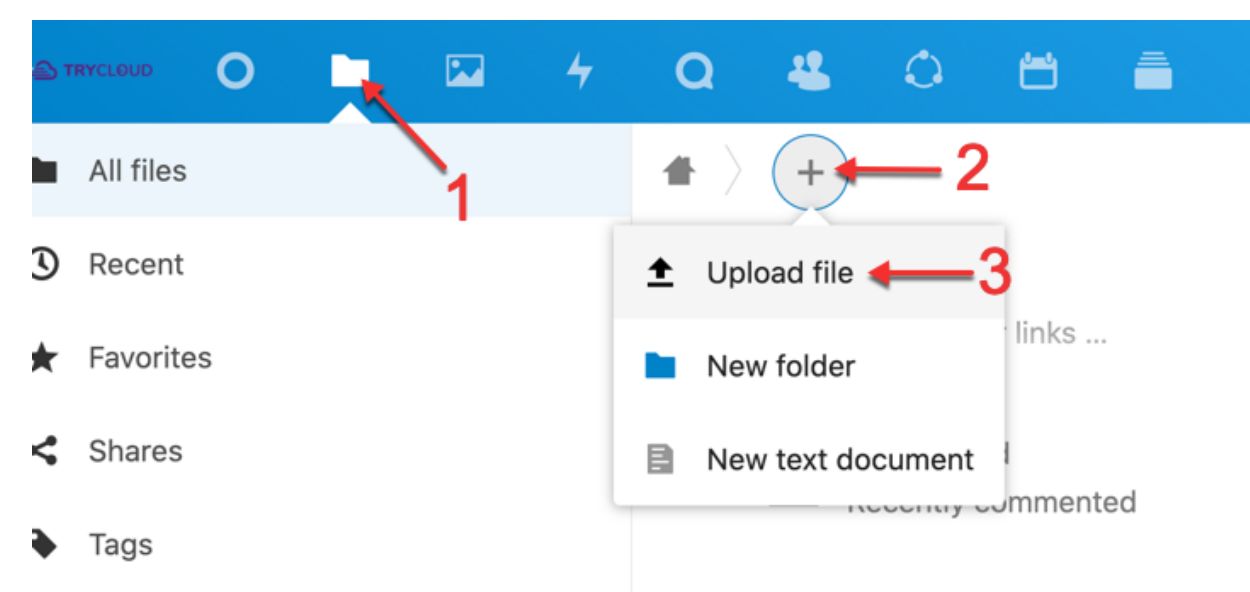
Given user on the dashboard page

When the user clicks the "Files" module

When the user clicks the add icon on the top

And users uploads file with the "upload file" option

Then verify the file is displayed on the page





7. **Story:** As a user, I should be able to manage folders.

**Feature:** As a user, I should be able to remove files from the favorites and upload a file directly

**Scenario:** Verify users can **add** the folder

Given user on the dashboard page

When the user clicks the "Files" module

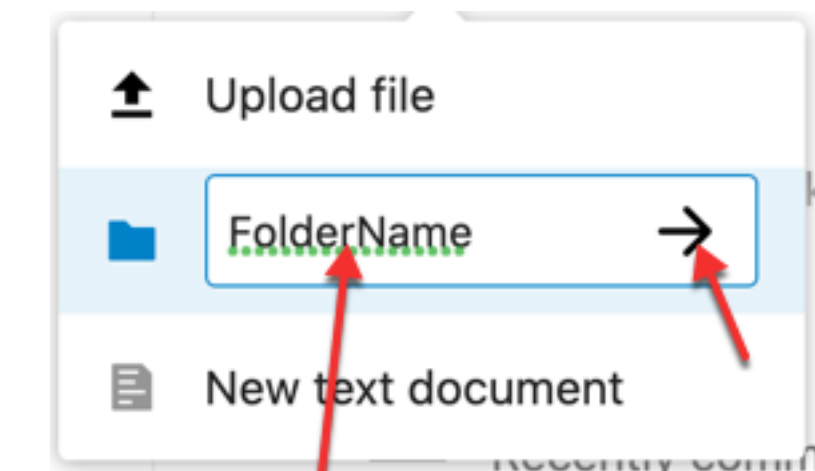
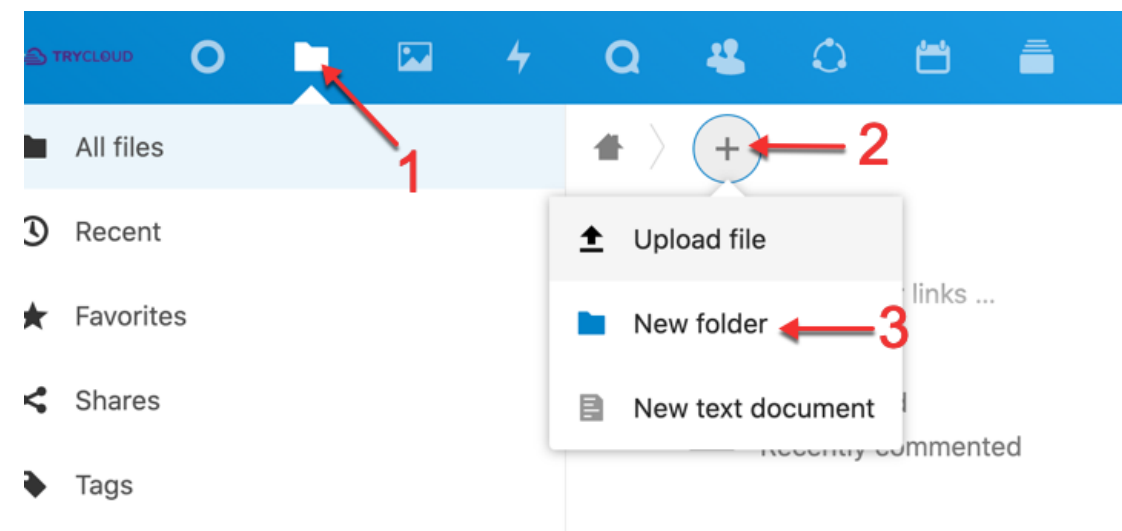
And user clicks the add icon on the top

And user click "new folder"

And user write a folder name

When the user click submit icon

Then Verify the folder is displayed on the page



**Scenario:** Verify users can **upload** a **file** inside a folder

Given user on the dashboard page

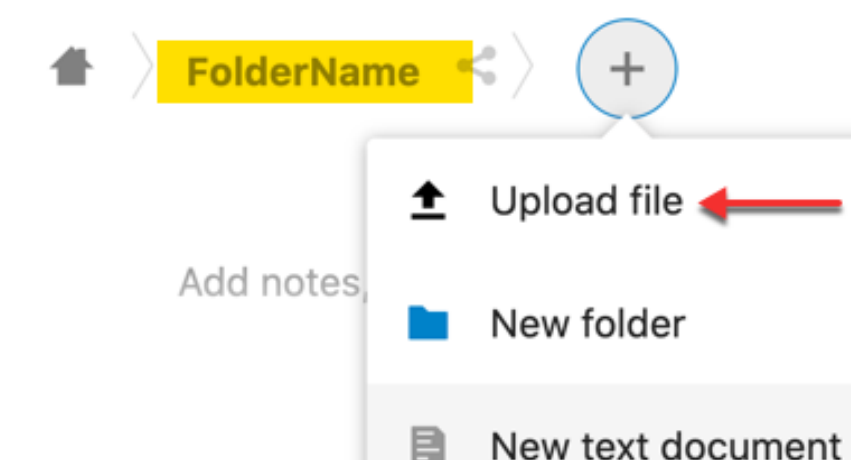
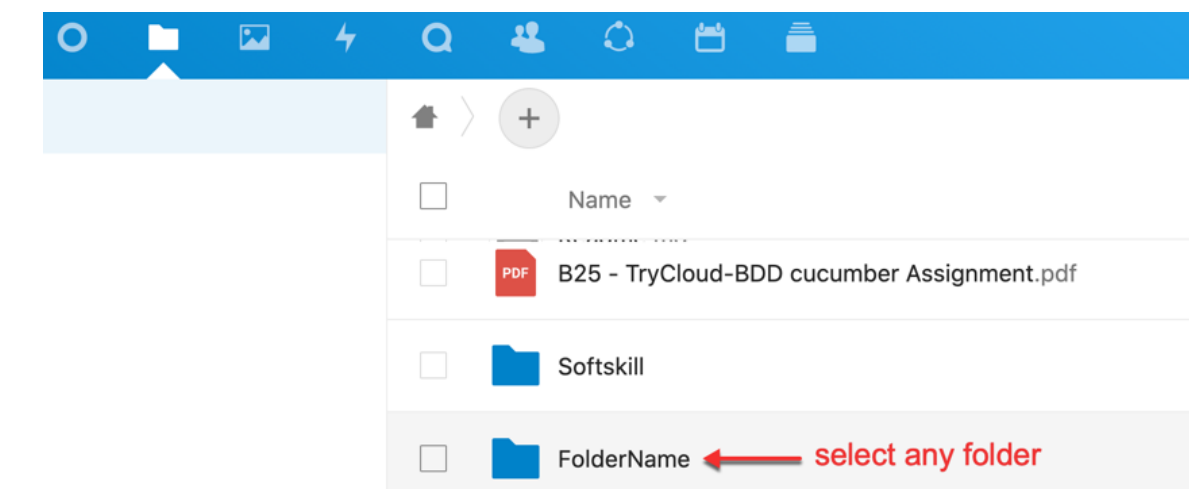
When the user clicks the "Files" module

And user choose a folder from the page

And user clicks the add icon on the top

When the user uploads a file with the upload file option

Then Verify the file is displayed on the page



8. **Story:** As a user, I should be able to delete a file/folder.

**Feature:** As a user, I should be able to delete a file/folder.

**Scenario:** Verify users **delete** a file/folder

Given user on the dashboard page

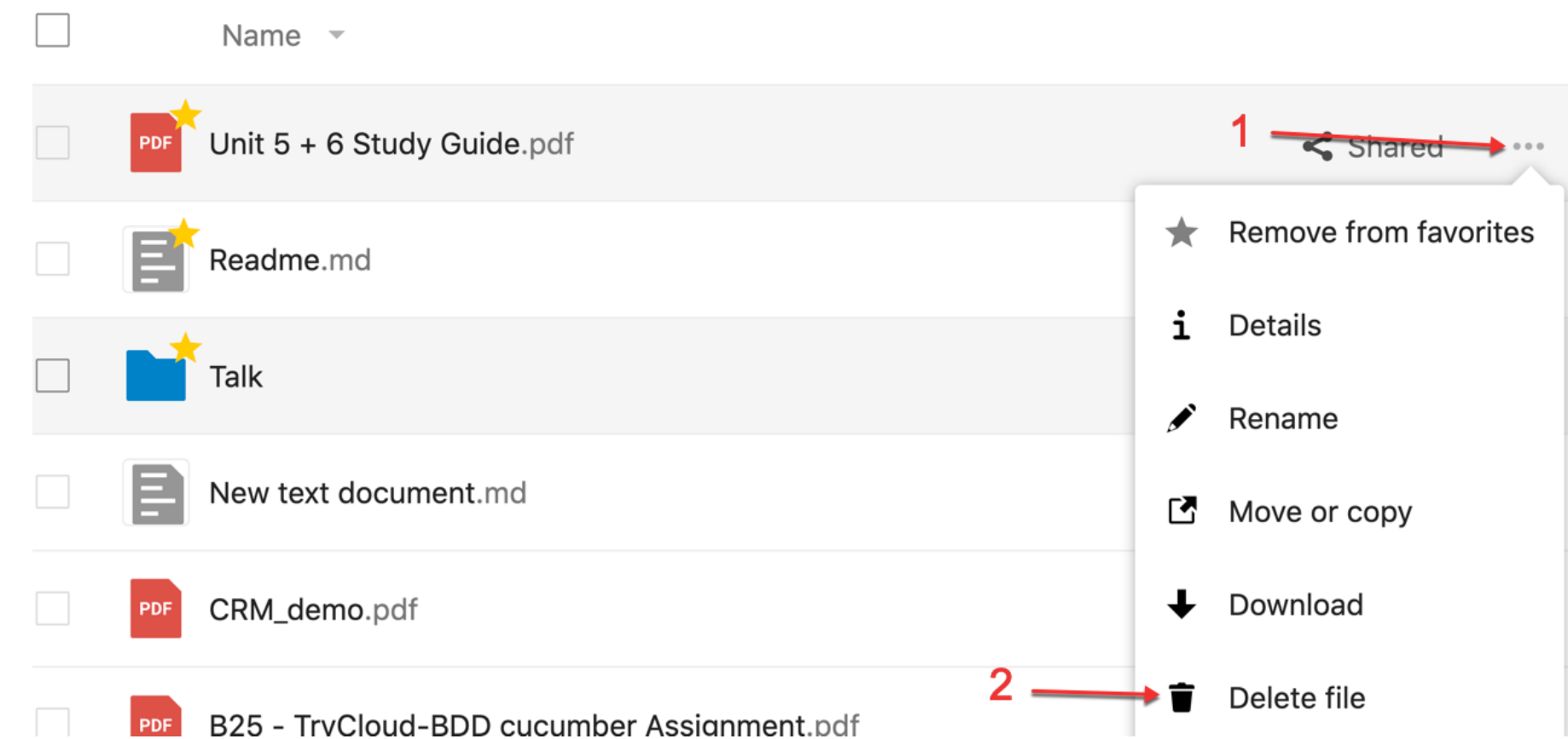
When the user clicks the "Files" module

And user click action-icon from any file on the page

And user choose the "Delete f" option

When the user clicks the "Deleted files" sub-module on the left side

Then Verify the deleted file is displayed on the page.



9. **Story:** As a user, I should be able to write comments to files/folders.

**Feature:** As a user, I should be able to write comments to files/folders.

**Scenario:** Verify users to write comments to files/folder

Given user on the dashboard page

When the user clicks the "Files" module

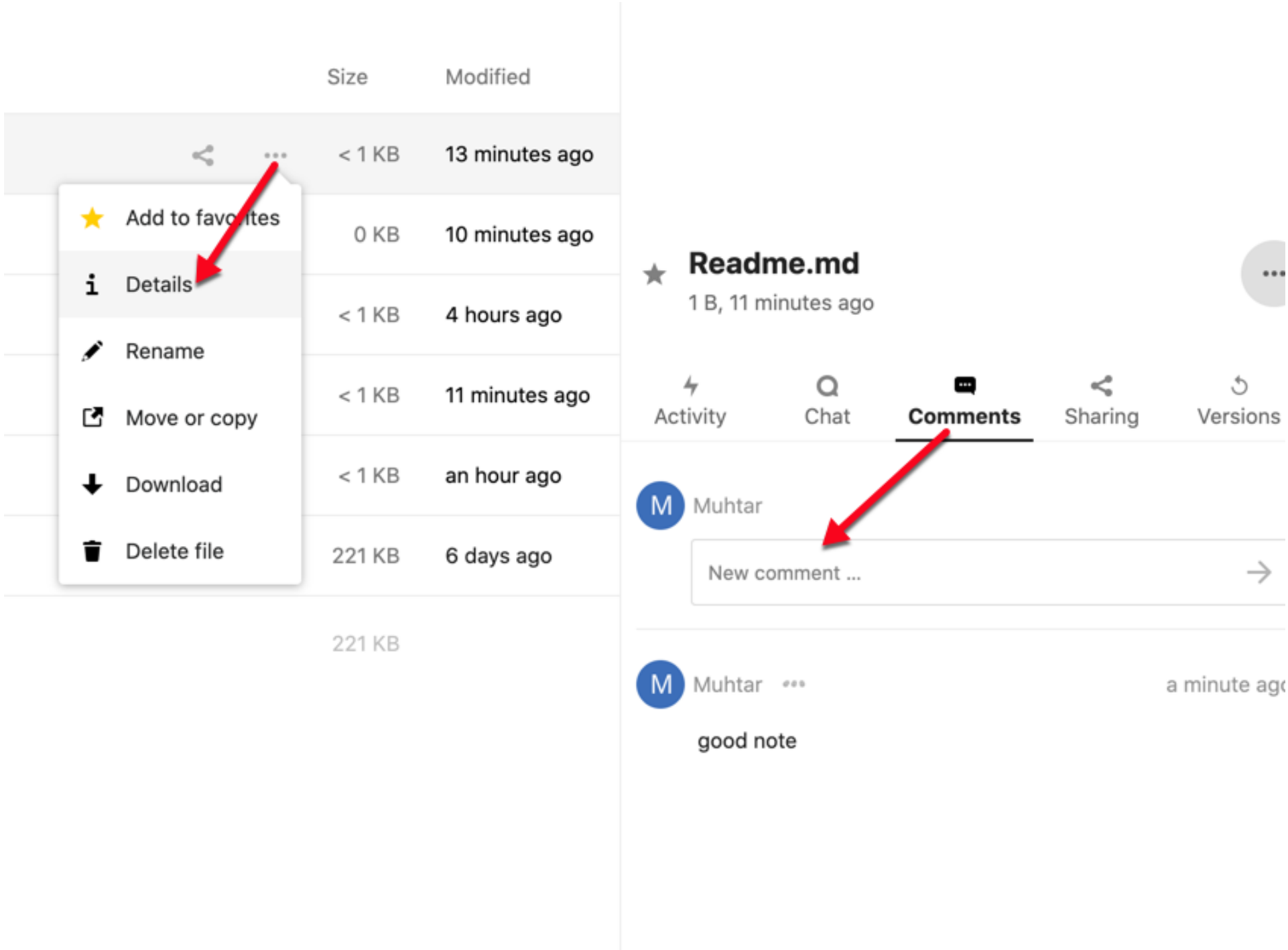
And user click action-icon from any file on the page

And user choose the "Details" option

And user write a comment inside the input box

And user click the submit button to post it

Then Verify the comment is displayed in the comment section.



10. **Story:** As a user, I should be able to update settings.

**Scenario:** Verify users update settings

Given user on the dashboard page

When the user clicks the "Files" module

And user clicks Settings on the left bottom corner

Then the user should be able to click any buttons

**Scenario:** Verify users to see the app storage usage

Given user on the dashboard page

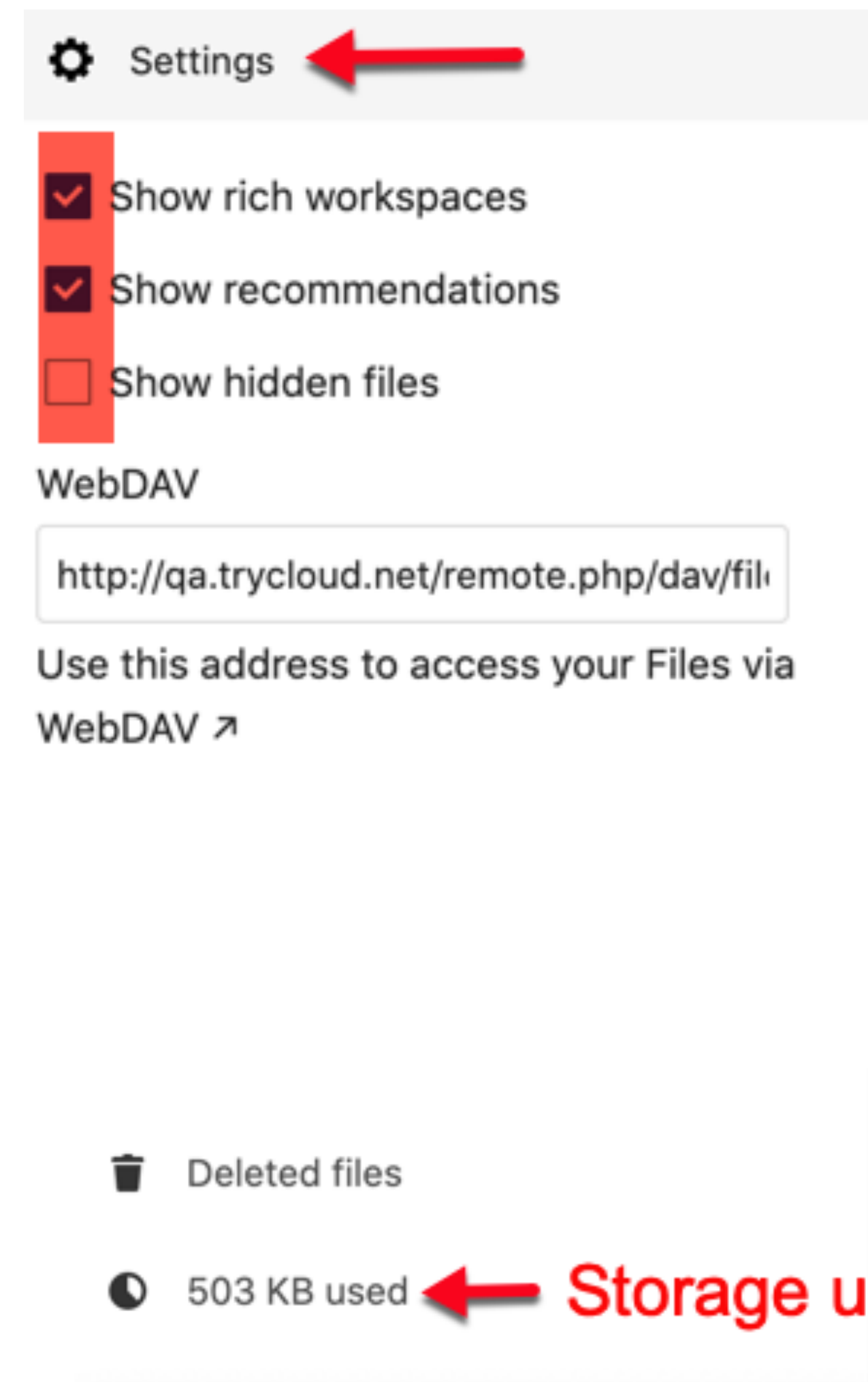
When the user clicks the "Files" module

And user checks the current storage usage

And user uploads file with the upload file option

And user refresh the page

Then the user should be able to see storage usage is increased



11. **Story:** As a user, I should be able to access to **Talks** module.

**Feature:** As a user, I should be able to access to Talks module

**Scenario:** verify users to access to Talks module

Given user on the dashboard page

When the user clicks the "Talk" module

Then verify the page title is "Talk - Trycloud QA"

**Scenario:** verify users to send a message

Given user on the dashboard page

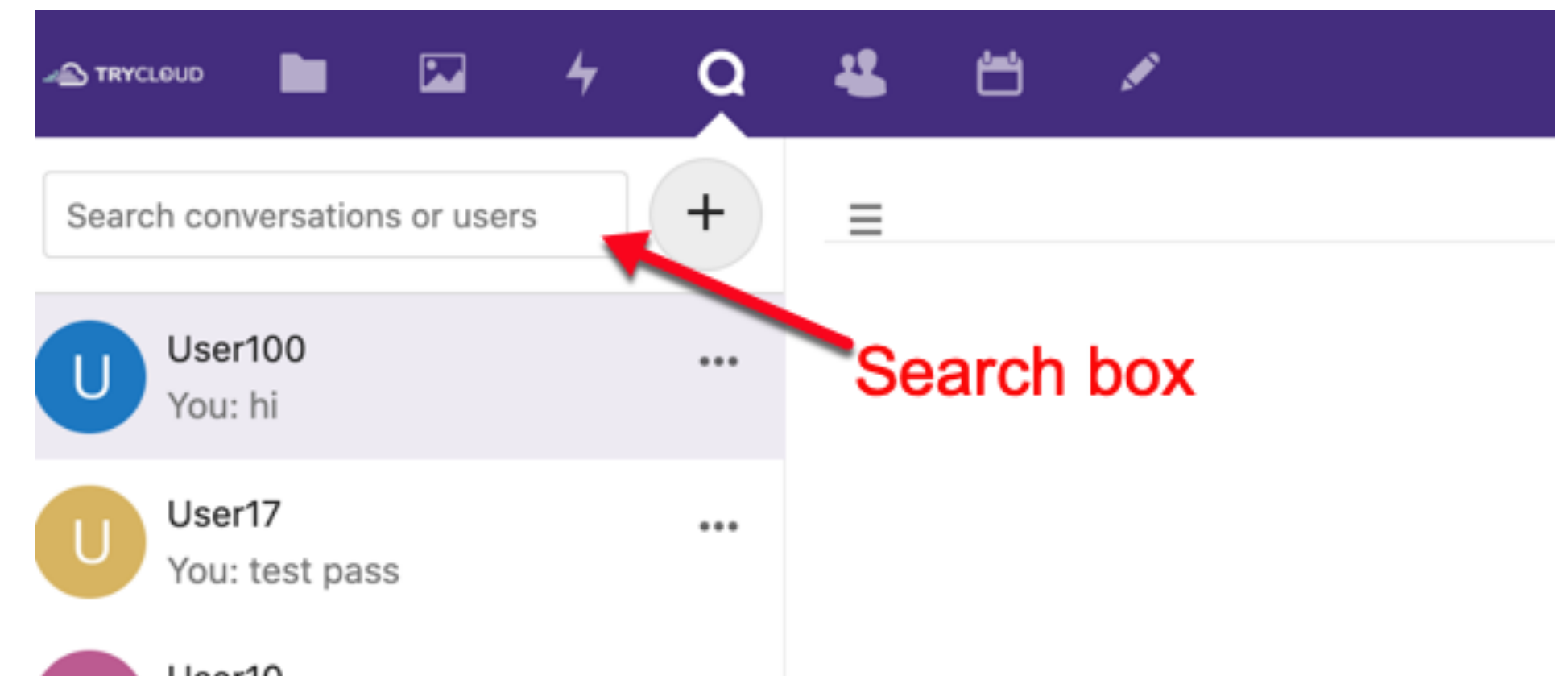
When the user clicks the "Talk" module

And user search user from the search box

And user write a message

And user clicks to submit button

Then the user should be able to see the message is displayed on the conversation log





12. **Story:** As a user, I should be able to access to **Contacts** module.

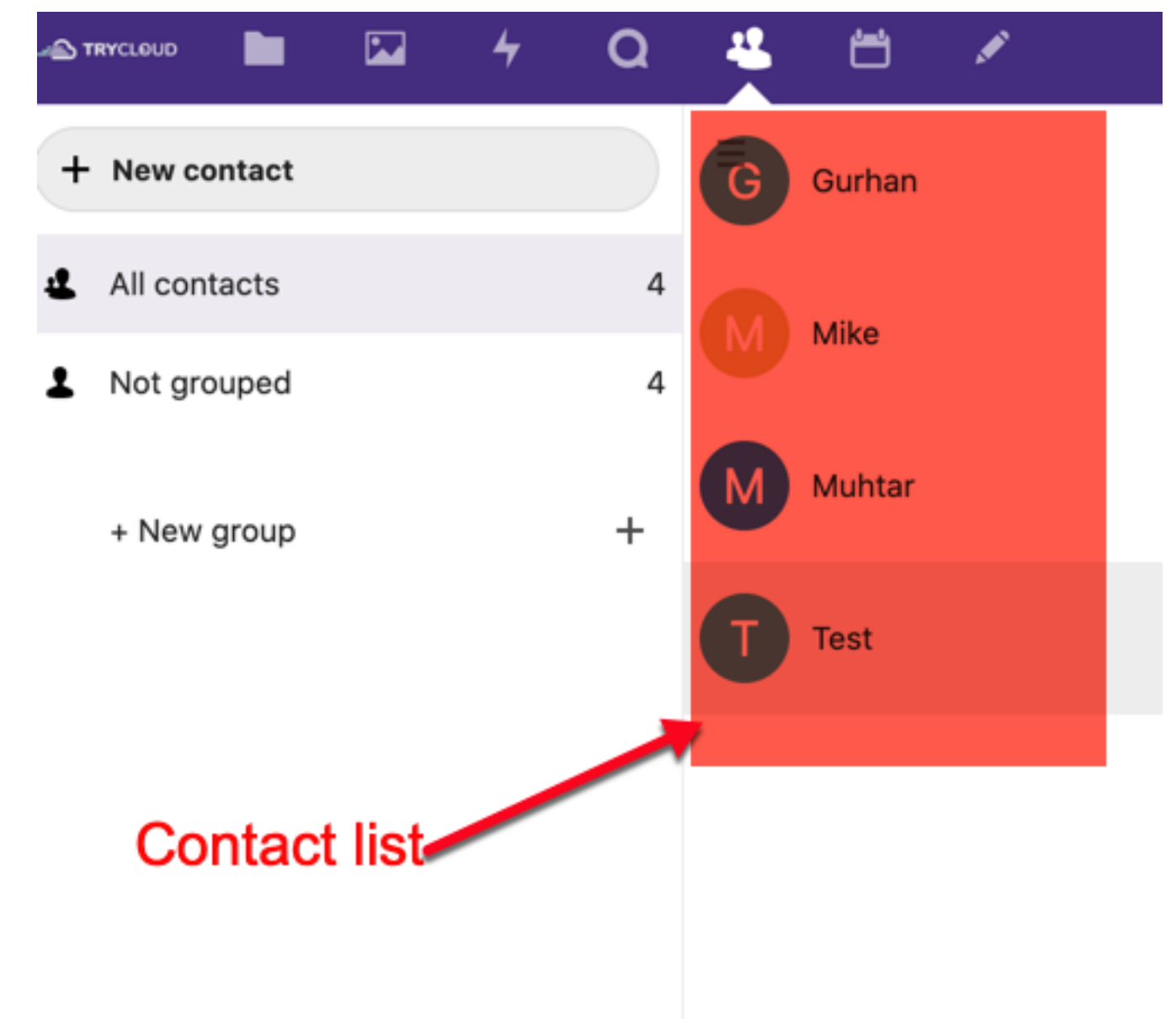
**Feature:** As a user, I should be able to access to Contacts module.

**Scenario:** verify user access to Talks module

Given user on the dashboard page

When the user clicks the "Talk" module

Then verify the page title is "Contacts - Trycloud QA"



13. **Story:** As a user, I should be able to view the contact list.

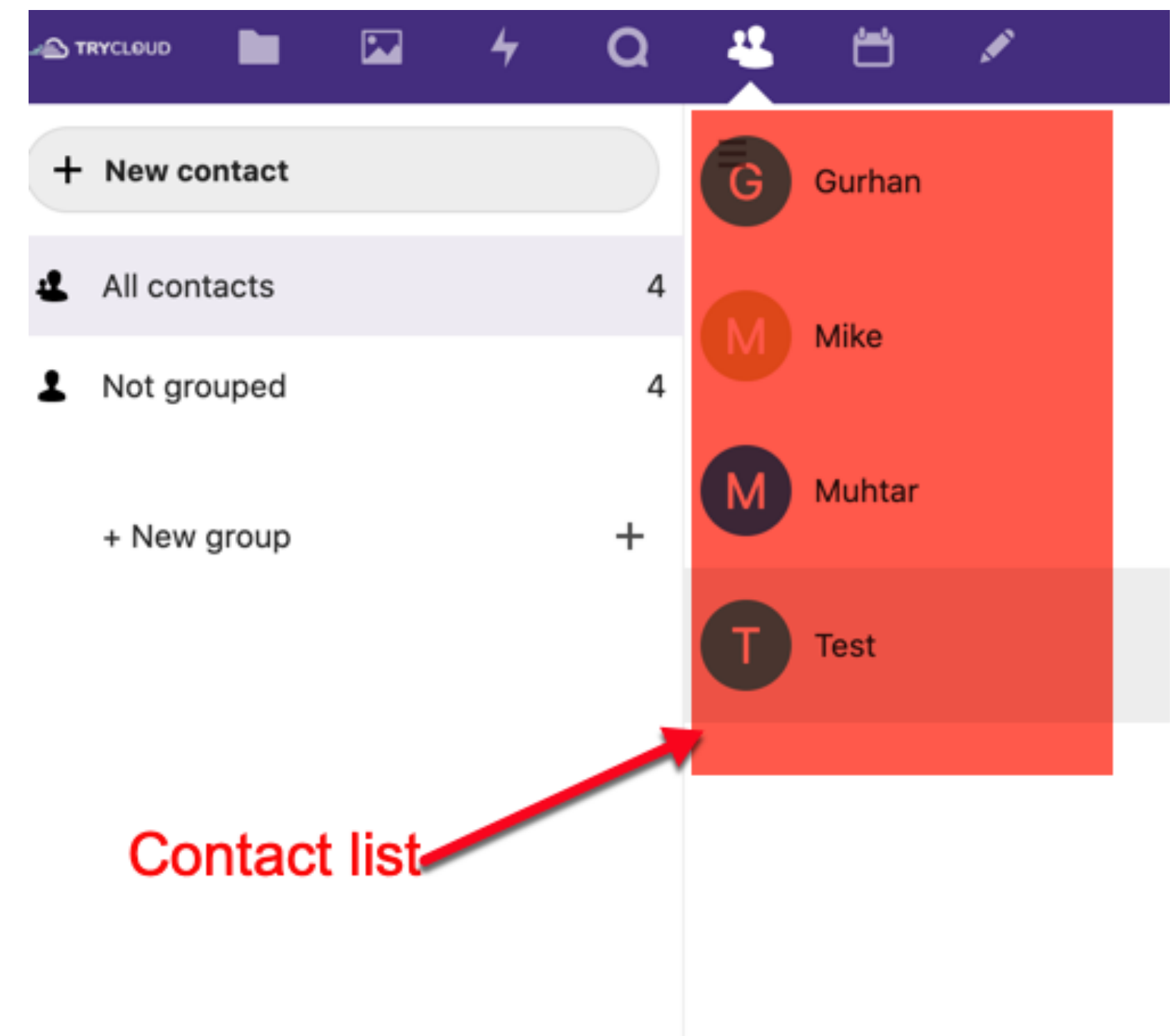
**Scenario:** verify users can see all the contact names on the contact list

Given user on the dashboard page

When the user clicks the contacts module

Then verify the contact names are in the list

(Pre-condition: there should be at least 2 contact names are displayed  
On the contact list so that view list function can be tested)



14. **Story:** As a user, I should be able to search any item/ users from the homepage.

**Feature:** As a user, I should be able to search any item/ users from the homepage.

**Scenario:** Verify users can search any files/folder/users from the search box.

Given user on the dashboard page

When the user clicks the magnifier icon 🔍 on the right top

And users search any existing file/folder/user name

Then verify the app displays the expected result option

